CLAIMS

1. A method for placing fast reroute backup tunnels between nodes of one or more pairs of nodes of a network to satisfy a requested total bandwidth for fast reroute backup tunnels between nodes of each said pair, said method comprising:

specifying a set of constraints on said backup tunnels; and

performing linear programming operations based on said set of constraints to find said backup tunnels, wherein the requested total bandwidth between nodes of each node pair may be divided among multiple backup tunnels.

2. The method of claim 1 wherein said set of constraints comprises:

for each said node pair including a first node and a second node:

a sum of backup tunnel bandwidths of backup tunnels exiting said first node should equal a requested total bandwidth of backup tunnels for said node pair;

a sum of backup tunnel bandwidths of backup tunnels entering said second node should equal said requested total bandwidth of backup tunnels for said node pair.

3. The method of claim 2 wherein said set of constraints further comprises:

for any link in said network, a sum of bandwidths consumed on the link by said backup tunnels does not exceed a backup bandwidth capacity of said link; and

for each node in said network other than nodes of said node pairs, a sum of bandwidths of backup tunnels entering such node should equal a sum of bandwidths of backup tunnels exiting such node.

4. The method of claim 3 wherein performing linear programming operations comprises:

employing an optimality function in addition to said set of constraints, said optimality function comprising a sum of bandwidth used on all links of said network in meeting said requested total bandwidth of each said node pair.

5. The method of claim 1 wherein said linear programming operations are performed on a network that does not include a protected node and links connected to said protected node.

6. The method of claim 1 wherein said linear programming operations are performed on a network that does not include a protected link.

7. The method of claim 1 wherein performing linear programming operations comprises:

finding links employed in said backup tunnels using said set of constraints and a linear programming procedure; and

identifying backup tunnels made up of said links.

8. The method of claim 5 wherein identifying backup tunnels made up of said links comprises:

establishing a subnetwork consisting of only links found by said linear programming procedure;

identifying a shortest path between said first node and said second node through said subnetwork;

identifying a link in said shortest path having a smallest remaining bandwidth;

eliminating said link having said smallest remaining bandwidth from said subnetwork;

reducing available bandwidth of links of said shortest path by the bandwidth of said link having said smallest remaining bandwidth; and

repeating establishing, identifying said shortest path, identifying said link in said shortest path, eliminating, and reducing until no links remain in said subnetwork.

9. A computer program product for placing fast reroute backup tunnels between nodes of one or more pairs of nodes of a network to satisfy a requested total bandwidth for fast reroute backup tunnels between nodes of each said pair, said computer program product comprising:

code that specifies a set of constraints on said backup tunnels;

code that performs linear programming operations based on said set of constraints to find said backup tunnels, wherein the requested total bandwidth between nodes of each node pair may be divided among multiple backup tunnels; and

a computer-readable storage medium that stores the codes.

10. The computer program product of claim 9 wherein said set of constraints comprises:

for each said node pair including a first node and a second node:

a sum of backup tunnel bandwidths of backup tunnels exiting said first node should equal a requested total bandwidth of backup tunnels for said node pair;

a sum of backup tunnel bandwidths of backup tunnels entering said second node should equal said requested total bandwidth of backup tunnels for said node pair.

11. The computer program product of claim 10 wherein said set of constraints further comprises:

for any link in said network, a sum of bandwidths consumed on the link by said backup tunnels does not exceed a backup bandwidth capacity of said link; and

for each node in said network other than nodes of said node pairs, a sum of bandwidths of backup tunnels entering such node should equal a sum of bandwidths of backup tunnels exiting such node.

12. The computer program product of claim 11 wherein said code that performs linear programming operations comprises:

code that employs an optimality function in addition to said set of constraints, said optimality function comprising a sum of bandwidth used on all links of said network in meeting said requested total bandwidth of each said node pair.

13. The computer program product of claim 9 wherein said linear programming operations are performed on a network that does not include a protected node and links connected to said protected node.

14. The computer program product of claim 9 wherein said linear programming operations are performed on a network that does not include a protected link.

15. The computer program product of claim 9 wherein said code that performs linear programming operations comprises:

code that finds links employed in said backup tunnels using said set of constraints and a linear programming procedure; and

code that identifies backup tunnels made up of said links.

16. The computer program product of claim 15 wherein said code that identifies backup tunnels made up of said links comprises:

code that establishes a subnetwork consisting of only links found by said linear programming procedure;

code that identifies a shortest path between said first node and said second node through said subnetwork;

code that identifies a link in said shortest path having a smallest remaining bandwidth;

code that eliminates said link having said smallest remaining bandwidth from said subnetwork;

code that reduces available bandwidth of links of said shortest path by the bandwidth of said link having said smallest remaining bandwidth; and

code that repeatedly invokes said code that identifies said shortest path, said code that identifies said link in said shortest path, said code that eliminates, and said code that reduces until no links remain in said subnetwork.

17. Apparatus for placing fast reroute backup tunnels between nodes of one or more pairs of nodes of a network to satisfy a requested total bandwidth for fast reroute backup tunnels between nodes of each said pair, said apparatus comprising:

a processor; and

a memory device storing instructions executed by said processor, said instructions comprising:

code that specifies a set of constraints on said backup tunnels;and

code that performs linear programming operations based on said set of constraints to find said backup tunnels, wherein the requested total bandwidth between nodes of each node pair may be divided among multiple backup tunnels.

18. The apparatus of claim 17 wherein said set of constraints comprises:

for each said node pair including a first node and a second node:

a sum of backup tunnel bandwidths of backup tunnels exiting said first node should equal a requested total bandwidth of backup tunnels for said node pair;

a sum of backup tunnel bandwidths of backup tunnels entering said second node should equal said requested total bandwidth of backup tunnels for said node pair.

19. The apparatus of claim 18 wherein said set of constraints further comprises:

for any link in said network, a sum of bandwidths consumed on the link by said backup tunnels does not exceed a backup bandwidth capacity of said link; and

for each node in said network other than nodes of said node pairs, a sum of bandwidths of backup tunnels entering such node should equal a sum of bandwidths of backup tunnels exiting such node.

20. The apparatus of claim 18 wherein said code that performs linear programming operations comprises:

code that employs an optimality function in addition to said set of constraints, said optimality function comprising a sum of bandwidth used on all links of said network in meeting said requested total bandwidth of each said node pair.

21. The apparatus of claim 17 wherein said linear programming operations are performed on a network that does not include a protected node and links connected to said protected node.

22. The apparatus of claim 17 wherein said linear programming operations are performed on a network that does not include a protected link.

23. The apparatus of claim 17 wherein said code that performs linear programming operations comprises:

code that finds links employed in said backup tunnels using said set of constraints and a linear programming procedure; and

code that identifies backup tunnels made up of said links.

24. The apparatus of claim 17 wherein said code that identifies backup tunnels made up of said links comprises:

code that establishes a subnetwork consisting of only links found by said linear programming procedure;

code that identifies a shortest path between said first node and said second node through said subnetwork;

code that identifies a link in said shortest path having a smallest remaining bandwidth;

code that eliminates said link having said smallest remaining bandwidth from said subnetwork;

code that reduces available bandwidth of links of said shortest path by the bandwidth of said link having said smallest remaining bandwidth; and

code that repeatedly invokes said code that identifies said shortest path, said code that identifies said link in said shortest path, said code that eliminates, and said code that reduces until no links remain in said subnetwork.

25. Apparatus for placing fast reroute backup tunnels between nodes of one or more pairs of nodes of a network to satisfy a requested total bandwidth for fast reroute backup tunnels between nodes of each said pair, said apparatus comprising:

means for specifying a set of constraints on said backup tunnels; and

means for performing linear programming operations based on said set of constraints to find said backup tunnels, wherein the requested total bandwidth between nodes of each node pair may be divided among multiple backup tunnels.